Optional Comma Separators in Numeric Data

in RiverWare 6.1 -- Overview and Implementation Notes

Author: Phil Weinstein, CADSWES

1.0 Overview

Original Task Description: *Provide an option in the configuration of every slot to display comma separators in numbers greater or equal to 1,000.*

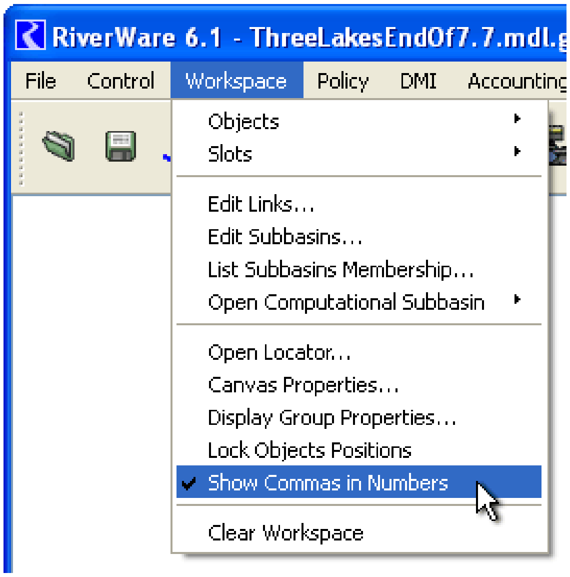
Instead of implementing this feature as a Slot configuration property (on a per-Slot basis), it was implemented as a global toggle in RiverWare.

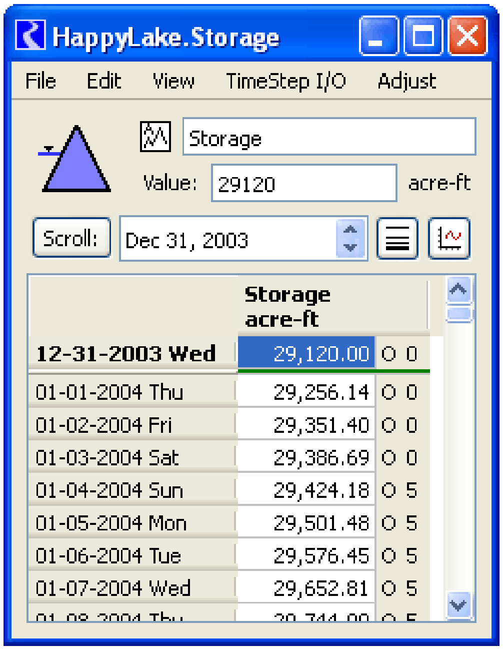
The application of this formatting option is currently specific to the display of numeric *Slot* data, but could readily be applied to *any* numeric display in RiverWare. (It just so happens that virtually all applicable numeric display in RiverWare is for Slot values).

Code to support the “Show Commas” toggle in multiple places was developed (initially in the View menu of all relevant Open Slot Dialogs and related Accounting Dialogs). But we ended up putting the checkbox in (only) the Workspace Window's Workspace Menu -- *see image.*

Dialogs supporting optional numeric comma display:

- Open Object Dialog (Slots Tab)
- Standard Open Slot Dialog (Series, Table and Periodic Slots)
- Open Scalar Slot Dialog
- Open Statistical Slot Dialog
- Edit Account Dialog
- Object Account Summary Dialog
- Exchange Balance Dialog
- SCT: Series Tab (table) and Scalar Tab (slot list)





Storage

Value: 29120 acre-ft

Scroll: Dec 31, 2003

	Storage acre-ft		
12-31-2003 Wed	29,120.00	O	0
01-01-2004 Thu	29,256.14	O	0
01-02-2004 Fri	29,351.40	O	0
01-03-2004 Sat	29,386.69	O	0
01-04-2004 Sun	29,424.18	O	5
01-05-2004 Mon	29,501.48	O	5
01-06-2004 Tue	29,576.45	O	5
01-07-2004 Wed	29,652.81	O	5

(Support in RPL numeric literal strings is being planned, but, as of 11-11-2010, has not been implemented).

The state of the global toggle persists in user-login based settings. On first use, i.e. starting a version of RiverWare having this feature for the first time, the default state is ON (i.e. *do* show commas). [Technical: Changes to the

"Show Commas in Numbers" toggle results in a WS_SHOW_NUMSEP_CHANGED callback issued from SimWS].

Numeric strings in data export files do not have commas. Numeric strings in text/HTML report files can have commas (when comma display is enabled), but this currently applies only to the DEMO HTML Simulation Object Report -- not available in the RiverWare release.

With the code changes to implement this feature, the distinction between strings used for display vs. strings used to initialize editors has been more explicitly characterized. Strings for editors lack commas and insignificant trailing zeros.

QLocale capabilities are currently used only for formatting strings with commas rather than for internationalization -- for now. Use of the QLocale is entirely encapsulated in Utils/cwFormat, and is currently hard-coded to use the Qt "c"-Locale (equivalent to English/US).

Entering commas while editing numeric values is not yet supported. This would need to be carefully, comprehensively implemented in all numeric editors. Furthermore, this should probably be done in conjunction with a careful analysis of support for international locales. An area of concern is that the European locales (generally speaking) REVERSE the meaning of commas and periods (for decimal points) -- with respect to those symbols used in the United States. Especially because of that, we would need to insure that all displays and editors are consistently supported.

2.0 Implementation Notes

The following implementation notes are provided specifically for the purpose of considering the decision to deploy this feature "early", in RiverWare 6.0.x. Changes have been put aside in a Solaris-based "workrel" area kept in synchronization with the "6.0 Prerel" CVS repository [as of 11-11-2010]:

- `Source: /projects/riverware/staff/philw/pwWorkRel/`
- `Object: /riverware/scratch/staff/philw/objWorkRel/`

Software Modifications to implement optional comma separators in displayed slot values, making use of QLocale numeric string formatting.

```
High Level Functional Areas:
  - Numeric value formatting.
  - Maintenance of the comma option toggle, with change notifications
  - GUI control of the comma option toggle, NOW LIMITED to the Workspace
  - GUI module mods to allow use of the new conditional numeric value
     formatting methods and to service toggle change notifications
     (to update their slot value displays).

Change overview is organized in these main sections, below:
  (1) LOW LEVEL FORMATTING UTILITIES, making use of the QLocale class.
  (2) TOGGLE CHANGE NOTIFICATION GENERATION
  (3) LOW-LEVEL SLOT-VALUE FORMATTING FUNCTIONS
  (4) GUI MODULE CHANGES
```

```
====================================================================
(1) LOW LEVEL FORMATTING UTILITIES, making use of the QLocale class.
====================================================================

------------------
Utils/cwFormat.hpp
Utils/cwFormat.cpp
------------------

New formatting functions were put into a new "cwFormat" name space, instead
of just having them in the global name space.

namespace cwFormat
{
   // Unconditionally Applied Locale-Based Formatting
   // Note: current support limited to SHOWING commas/numeric separators.
   QString formatLocale (double value, const QString& fmt, int precision);
   QString formatLocaleFullPrec (double value, const QString& fmt);

   // Conditionally Applied Locale-Based Formatting
   // Note: currently used to switch between showing/hiding numeric separators.
   QString formatConditionalLocale (double val, const QString& fmt, int prec);
   QString formatConditionalLocaleFullPrec (double val, const QString& fmt);

   // Switch for showing (or not) numeric separator
   // On user change, SimWS WS_SHOW_NUMSEP_CHANGED callback is issued.
   void setShowNumSeparator (bool);   // sets static flag
   bool showNumSeparatorEna();

   // Numeric separator information (computed from current locale)
   QString numericSeparatorName (bool plural);
   QChar numericSeparator();
}

=========================================
(2) TOGGLE CHANGE NOTIFICATION GENERATION
=========================================

--------------------
Sim/CallbackType.hpp
Sim/CallbackType.cpp
Sim/SimWS.cpp
Sim/SimWS.hpp
--------------------

Notifications for change in the global "show numeric separator" toggle.

New Callback Type:
  WS_SHOW_NUMSEP_CHANGED, // show number separator global flag changed

New method to issue the WS_SHOW_NUMSEP_CHANGED callback from SimWS:
  void notifyShowNumberSeparatorFlagChanged();

=============================================
(3) LOW-LEVEL SLOT-VALUE FORMATTING FUNCTIONS
=============================================

--------------------
Sim/SlotGUIUtils.hpp
Sim/SlotGUIUtils.cpp
--------------------

Added optional 'forEdit' parameter to these methods; default: false.

 QString formatValueUsrStrForUnitScope (const SeriesSlot*, bool forAccum,
                                        double value,
                                        bool usrPrecision=true,
                                        bool forEdit=false);

 QString formatValueUsrStr (const Slot*, int colInx, double value,
                            bool usrPrecision=true,
                            bool forEdit=false);

Added non-optional 'forEdit' parameter to this method which also supports
"alt units" (for Accounting Slots):

 QString getValueUsrStr (const Slot*, int slotTinx, int colInx,
                         bool forEdit, bool altUnits);

------------------
Sim/RootColRef.hpp (SlotColRef)
Sim/RootColRef.cpp (SlotColRef)
------------------

Added 'forEdit' parameter to these methods, non-optional:

  QString formatUserPrimValue (
     double userPrimVal, bool showNaN, bool forEdit) const;
  QString formatUserAltValue (
     double userAltVal, bool showNaN, bool forEdit) const;
  QString getUserPrimValueStr (
     const Date_Time&, bool showNaN, bool forEdit) const;
  QString getUserAltValueStr (
     const Date_Time&, bool showNaN, bool forEdit) const;

======================
(4) GUI MODULE CHANGES
======================

-------------------
Q3GUI/Workspace.hpp
Q3GUI/Workspace.cpp
-------------------

(A) Implementation of the "Show Comma in Numbers" checkbox in the Workspace
 menu:

New data members:
  QAction* _showCommasInNumAction; -- menu item in the Workspace menu.

New methods:
  void aboutToShow_WorkspaceMenu(); -- to set the state of the checkbox.
  void showCommasInNum_toggled (bool); -- to respond to checkbox changes/

(B) QSettings-based persistence of the global toggle state:

  // QSettings Keys
  static const char* SHOW_NUM_SEPS_ENA   ("ShowNumberSepsEna");

  See changes in methods:
   void Workspace::loadWorkspaceQSettings()
   void Workspace::saveWorkspaceQSettings()

  Receiving notifications:
   WS_SHOW_NUMSEP_CHANGED callback handled in method Workspace::
     callbackHandler (CallbackType, CallbackData*) to peform
     saveWorkspaceQSettings().

---------------------------------
Q3GUI/OpenObjectDlg.hpp
Q3GUI/OpenObjectDlg.cpp
Q3GUI/OpenObjectDlg.ListItems.cpp
---------------------------------

Added callback handler for callbacks from SimWS:

  Callback* _simWsCallback;
  int simWsCallbackHandler (CallbackType, CallbackData*, void*);
     WS_SHOW_NUMSEP_CHANGED callback: call updateSlotItems();

  See also additions in:
    void OpenObjectDlg::addCallbacks()
    void OpenObjectDlg::deleteCallbacks()

-----------------------
Q3GUI/ScalarSlotDlg.hpp
Q3GUI/ScalarSlotDlg.cpp
-----------------------

Added callback handler for callbacks from SimWS:

  Callback* _simWsCallback;
  int simWsCallbackHandler (CallbackType, CallbackData*, void*);
    WS_SHOW_NUMSEP_CHANGED callback: call updateSlotDisplayValueLabel();

  See also additions in:
    void ScalarSlotDlg::addCallbacks()
    void ScalarSlotDlg::deleteCallbacks()

Added method:
  QString formatUserValForEdit (double userVal) const;
```

```
------------------------
Q3GUI/SlotDataQTable.hpp
Q3GUI/SlotDataQTable.cpp
Q3GUI/SlotQDlgTable.hpp
Q3GUI/SlotQDlgTable.cpp
QtSCT/SctQTable.hpp
QtSCT/SctQTable.cpp
------------------------

Added 'forEdit' parameter to virtual SlotDataQTable method:

  virtual QString cellValStr (int row, int col,
                              bool forEdit, bool fullPrec) const = 0;

  ... implementations of the two subclasss are used in the base-class
  implementation of SlotDataQTable::paintNumFlagCell (QPainter* ...).

Changes in methods:
  void SlotQDlgTable::refreshColumnMapLabels()
  QString SlotQDlgTable::cellSeriesSlotValStr (int row, int col, ...
  QString SlotQDlgTable::cellFlags (int row, int col) const
  void SlotQDlgTable::copyCellDataToMainEdit (int row, int col)
  int SlotQDlgTable::columnDataContentWidth_timeCols (int col)
  int SlotQDlgTable::columnDataContentWidth_slotCols (int col)
  QWidget* SlotQDlgTable::createEditor (int row, int col, ...
  QString SlotQDlgTable::cellGridSelectionStr() const

-------------------
Q3GUI/SlotQtDlg.hpp
Q3GUI/SlotQtDlg.cpp
-------------------

In callback handler: SlotQtDlg::simWsCallbackHandler (CallbackType ...
  WS_SHOW_NUMSEP_CHANGED ... updates to the SlotQDlgTable and status bar.

---------------------------
QtSCT/SctModelData.hpp
QtSCT/SctModelData.Sim.cpp
---------------------------

Condensed two discrete SimWS callbacks (registrations) into a single general
SimWS callback to process the new callback type:

Removed:
  Callback *_rwObjDispatchChangedCallback;
  Callback *_rwObjRenamedCallback;
  int rwWorkspaceCallbackHandler (CallbackType, CallbackData*, void*);

Added:
  Callback* _simWsCallback;
  int simWsCallbackHandler (CallbackType, CallbackData*, void*);
    WS_SHOW_NUMSEP_CHANGED: send update requests to "listener" client.

Also, changes in:
  okstat SctModelData::getSlotScaleUnitsDisplayStr (int slotInx, ...
  okstat SctModelData::formatSlotScaleUnitValue (int slotInx,

---------------------------
QtSCT/SctView.Selection.cpp
---------------------------

Minor comment that the new formatting functions are NOT used here.

----------------------------
QtSCT/SctDialog.Handlers.cpp
----------------------------

Change in method: SctDialog::scalarListPanel_slotSelectionChanged (...)
  ... in computation of strings for MainLineEditor and Unit Label.

--------------------------
Q3GUI/SlotConfigQtDlg.cpp
QtUtils/RepGenSlot.cpp
QtUtils/RepGenSlotHtml.cpp
QtUtils/RepGenSlotText.cpp
-------------------------

Client modifications for SlotGUIUtils formatting method parameter changes.

-------------------------
QtUtils/SlotListPanel.hpp
QtUtils/SlotListPanel.cpp
-------------------------

Added mechanism to refresh list, preserving the item selection, and to
update the user value field in items:

New methods:
  void SlotListPanel::rebuildListView()
  void SlotListPanel::setSelectedSlotList (const cwSlist<SlotColRef>&, ...
  void SlotListPanel::SlotListView::setSelectedSlotColRefs (const ...
  void SlotListPanel::SlotListViewItem::setUserValField (const QString&);
  QString SlotListPanel::SlotListViewItem::editValStr()
  QString SlotListPanel::SlotListViewItem::formatValForEdit (double val)

New data members, used to prevent feedback looking:
  bool SlotListPanel::SlotListViewItem::_setValue_inProgress;

In callback handler: SlotListPanel::rwWorkspaceCallbackHandler (CallbackType..
  WS_SHOW_NUMSEP_CHANGED ... call rebuildListView.

Changes in:
  void SlotListPanel::slotListView_doubleClicked (QTreeWidgetItem*)
  void SlotListPanel::slotListView_contextAboutToShow()
  void SlotListPanel::context_EditSlotValue_activated()
  void SlotListPanel::context_CopySlots_activated()
  void SlotListPanel::SlotListViewItem::SlotListViewItem constructor
  void SlotListPanel::SlotListViewItem::refreshUserVal()
  void SlotListPanel::SlotListViewItem::startRename (int col)
  void SlotListPanel::SlotListViewItem::processCellEdit (int col)

Unrelated fix: reconstrain icon column widths after calling "fit" methods"
  void SlotListPanel::fitPanelColumnsAll (bool growOnly /*=false*/)
  void SlotListPanel::fitPanelColumnsData (bool growOnly /*=false*/)
  void SlotListPanel::fitPanelColumnsHeader (bool growOnly /*=false*/)

--------------------------
Q3GUI/StatTableSlotDlg.hpp
Q3GUI/StatTableSlotDlg.cpp
--------------------------

Numeric text now uses the conditional comma formatting methods.

  Changes in:
    void StatTableSlotDlg::updateTableValues()
    void StatTableSlotDlg::SlotTable::paintCell (QPainter *painter, ...

Added callbacks from SimWS to service change to the "show commas" toggle.

  Added:
    Callback* _simWsCallback;
    int simWsCallbackHandler (CallbackType, CallbackData*, void*);
    .. WS_SHOW_NUMSEP_CHANGED .. updateTableValues and updateTableLabels.

---
```